



**TUGAS AKHIR - TE141599**

**DESAIN SISTEM KONTROL OPTIMAL *LINEAR*  
*QUADRATIC GAUSSIAN* UNTUK SISTEM *MIMO* PADA  
*BOILER PLANT SIMULATOR***

Yudi Galih Permana  
NRP 2212100085

Dosen Pembimbing  
Eka Iskandar, S.T., M.T.  
Ir. Rusdhianto Effendi A.K., M.T.

JURUSAN TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2016



**FINAL PROJECT - TE141599**

**DESIGN OPTIMAL CONTROL SYSTEM FOR MIMO  
SYSTEM ON BOILER PLANT SIMULATOR**

Yudi Galih Permana  
NRP 2212100085

Supervisor  
Eka Iskandar, S.T., M.T.  
Ir. Rusdhianto Effendi A.K., M.T.

DEPARTEMENT OF ELECTRICAL ENGINEERING  
Faculty of Industrial Technology  
Sepuluh Nopember Institut of Technology  
Surabaya 2016

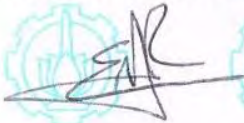
**DESAIN SISTEM KONTROL OPTIMAL LINEAR QUADRATIC  
GAUSSIAN UNTUK SISTEM MIMO PADA *BOILER PLANT*  
SIMULATOR**

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik  
Pada  
Bidang Studi Teknik Sistem Pengaturan  
Jurusan Teknik Elektro  
Institut Teknologi Sepuluh Nopember**

**Menyetujui,**

**Dosen Pembimbing I**



**Eka Iskandar, S.T., M.T.**

**NIP. 1980 05 28 2008 12 1001**

**Dosen Pembimbing II**



**Ir. Rusdhianto Effendi A.K., M.T.**

**NIP. 1957 04 24 1985 02 1001**



**SURABAYA  
JUNI, 2016**

## **Desain Sistem Kontrol Optimal *Linear Quadratic Gaussian* Untuk Sistem MIMO Pada *Boiler Plant Simulator***

Nama : Yudi Galih Permana  
Pembimbing : Eka Iskandar, S.T., M.T.  
Ir. Rusdhianto Effendi A.K., M.T.

### **ABSTRAK**

Dalam industri *power plant*, perubahan beban yang terjadi karena permintaan daya akan menyebabkan perubahan variabel proses yang lain, misalnya tekanan, ketinggian air, dan temperatur. Padahal variabel – variabel proses tersebut harus dijaga nilainya agar tetap konstan pada nilai tertentu meskipun terjadi perubahan beban. Untuk mengatasi hal tersebut, dalam tugas akhir ini akan dibahas perancangan kontroler yang mampu mengatasi perubahan beban sehingga tidak mengakibatkan perubahan yang besar pada variabel proses yang lain. Metode yang digunakan yaitu kontrol optimal *Linear Quadratic Gaussian*. Metode kontrol optimal *Linear Quadratic Gaussian* adalah gabungan dari filter kalman dan *Linear Quadratic Regulator* (LQR). Metode kontrol LQG ini bisa menangani perubahan beban dan mengembalikan *state* kembali ke *operating point* ketika terjadi perubahan beban.

**Kata kunci:** *Linear Quadratic Gaussian, nonlinear boiler-turbine, sistem multivariable*

***Design Optimal Control System Linear Quadratic Gaussian for  
MIMO System on Boiler Plant Simulator***

*Name* : Yudi Galih Permana  
*Advisor* : Eka Iskandar, S.T., M.T.  
Ir. Rusdhianto Effendi A.K., M.T.

***ABSTRACT***

In the power plant industry, load changes occurs because the demand for power will lead to changes process variable, such as pressure, water level, and temperature. Whereas the value of process variables must be maintained in order to be constant despite of load changes. To solve this problem, in this final project will discuss the design of controller that can handle the load changes so that the load changes do not affect in the change of the value of process variables. The method used in this final project is optimal controller Linear Quadratic Gaussian. Linear Quadratic Gaussian is a combination of Kalman Filter and Linear Quadratic Regulator. This method of LQG can handle the load changes and regulate the state to the operating point.

***Keywords*** : *Linear Quadratic Gaussian, nonlinear boiler-turbine, multivariable system*

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>PERNYATAAN KEASLIAN TUGAS AKHIR.....</b>	<b>v</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>vii</b>
<b>ABSTRAK .....</b>	<b>ix</b>
<b>ABSTRACT .....</b>	<b>xi</b>
<b>KATA PENGANTAR.....</b>	<b>xiii</b>
<b>DAFTAR ISI.....</b>	<b>xv</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	1
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	2
1.5 Metodologi.....	2
1.6 Sistematika Penulisan .....	4
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>7</b>
2.1 <i>Boiler-Turbine</i> .....	7
2.2 Model Matematika <i>Boiler-Turbine</i> .....	8
2.3 Linierisasi .....	10
2.4 Sistem Kontrol Optimal .....	13
2.4.1 Kontrol Optimal <i>Linear Quadratic Gaussian</i> .....	15
2.4.2 <i>Controllability</i> .....	17
2.4.3 <i>Observability</i> .....	17
2.5 Wonderware Intouch.....	17
<b>BAB III PERANCANGAN SISTEM.....</b>	<b>21</b>
3.1 Linierisasi Model Matematika <i>Boiler-Turbine</i> .....	21
3.2 Pengujian <i>Controllability</i> dan <i>Observability</i> .....	25
3.3 Perancangan Kontroler .....	26
3.3.1 Perancangan Estimator.....	29
3.4 Perancangan Simulasi <i>Boiler-Turbine</i> .....	33
3.5 Perancangan <i>Human Machine Interface</i> .....	36
<b>BAB IV PENGUJIAN DAN ANALISIS DATA .....</b>	<b>39</b>
4.1 Simulasi .....	39
4.1.1 Simulasi Menggunakan LQR .....	39

4.1.2 Simulasi Menggunakan LQG.....	40
<b>BAB V PENUTUP .....</b>	<b>47</b>
5.1 Kesimpulan.....	47
5.2 Saran.....	47
<b>DAFTAR PUSTAKA.....</b>	<b>49</b>
<b>LAMPIRAN .....</b>	<b>51</b>
Program <i>Linear Quadratic Gaussian</i> pada Matlab .....	51
<b>RIWAYAT PENULIS .....</b>	<b>53</b>

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Diagram Sederhana dari <i>Boiler-Turbine Unit</i> .....	7
<b>Gambar 2.2</b> Linierisasi $y=f(x)$ di Sekitar $x^*$ .....	10
<b>Gambar 2.3</b> Diagram Blok LQG .....	16
<b>Gambar 2.4</b> Tampilan <i>Window Maker</i> Pada <i>Software Wonderware</i> Intouch .....	19
<b>Gambar 3.1</b> Diagram Blok LQG Regulator.....	26
<b>Gambar 3.2</b> Diagram Blok Filter Kalman .....	30
<b>Gambar 3.3</b> Diagram Blok Simulasi di Simulink .....	33
<b>Gambar 3.4</b> Diagram Blok Subsistem <i>Nonlinear Boiler-Turbine</i> .....	34
<b>Gambar 3.5</b> Diagram Blok Subsistem Filter Kalman .....	35
<b>Gambar 3.6</b> Diagram Blok Subsistem Sinyal Referensi.....	35
<b>Gambar 3.7</b> Diagram Blok Subsistem U Nominal .....	36
<b>Gambar 3.8</b> Tampilan <i>Human Machine Interface</i> .....	37
<b>Gambar 4.1</b> Output Sistem Dengan <i>Noise</i> 5% Menggunakan LQR ....	40
<b>Gambar 4.2</b> Output Sistem Dengan <i>Noise</i> 5% Dari Titik Kerja .....	41
<b>Gambar 4.3</b> Output Sistem Dengan <i>Noise</i> 10% Dari Titik Kerja .....	41
<b>Gambar 4.4</b> Sinyal Kontrol Untuk Sistem Dengan <i>Noise</i> 5% Dari titik Kerja.....	42
<b>Gambar 4.5</b> Sinyal Kontrol Untuk Sistem Dengan <i>Noise</i> 10% Dari titik Kerja.....	43
<b>Gambar 4.6</b> Output Sistem Dengan <i>Noise</i> 5% Dari titik Kerja Dan Perubahan Beban Pada Steam 0,01 .....	44
<b>Gambar 4.7</b> Output Sistem Dengan <i>Noise</i> 10% Dari titik Kerja Dan Perubahan Beban Pada Steam 0,01 .....	44



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*Boiler-turbine* adalah suatu Pembangkit Listrik Tenaga Uap (PLTU) dalam skala kecil, yang merupakan suatu sistem pembangkit tenaga listrik dengan mengubah energi kimia dengan bahan bakar seperti batu bara, minyak bumi, ataupun gas bumi menjadi energi mekanik, dengan memanfaatkan energi kinetik uap (*steam*) untuk menggerakkan turbin. Tujuan *boiler-turbine* adalah menghasilkan uap (*steam*) yang berasal dari air sungai atau laut yang diproses *demineralized water* terlebih dahulu untuk dipanaskan ke dalam *boiler* sehingga menjadi uap jenuh. Untuk menggerakkan turbin diperlukan uap panas lanjut (uap kering) agar tidak merusak turbin. Alat yang dapat mengubah uap basah menjadi uap kering disebut *superheater*. Uap yang keluar dari *superheater* mempunyai tekanan yang sama dengan keluaran *boiler*, dan mempunyai temperatur lebih tinggi dari *boiler* [1].

Pemodelan *boiler-turbine* termasuk kategori sistem *nonlinear* yang memiliki banyak ketidakpastian. Ketidakpastian tersebut dapat berupa gangguan eksternal, ketidakpastian model, variasi parameter, ataupun *error* yang muncul pada saat linierisasi. Ketidakpastian-ketidakpastian ini dapat mempengaruhi kestabilan sistem jika tidak diantisipasi oleh sistem kontrol. Selain itu, pemodelan *boiler-turbine* menggunakan konfigurasi sistem *Multi Input Multi Output* (MIMO) [2].

Permasalahan yang terjadi pada *boiler-turbine* adalah terdapat ketidakpastian dan pemodelan *boiler-turbine* menggunakan konfigurasi sistem *Multi Input Multi Output*. Sehingga variabel yang diatur pada *boiler-turbine* mengalami kesulitan mengikuti sinyal referensi berupa beberapa kondisi titik operasi yang diinginkan [3].

Untuk mengatasi permasalahan tersebut, pada Tugas Akhir ini akan dibuat desain sistem kontrol optimal *Linear Quadratic Gaussian* (LQG) untuk mengatur konfigurasi plant MIMO *boiler-turbine*.

### 1.2 Perumusan Masalah

Perumusan masalah yang dibahas dalam Tugas Akhir ini adalah terdapat ketidakpastian konfigurasi sistem *Multi Input Multi Output* (MIMO) pada *boiler-turbine*. Sehingga variabel yang diatur pada *boiler-turbine* mengalami kesulitan mengikuti sinyal referensi berupa kondisi

titik operasi yang diinginkan. Untuk mengatasi masalah tersebut, dirancang metode control optimal *Linear Quadratic Gaussian* untuk mengatasi konfigurasi plant MIMO pada *boiler-turbine*.

### 1.3 Tujuan

Dengan adanya permasalahan yang timbul, maka penelitian yang dilakukan adalah dengan merancang sistem kontrol, yaitu kontrol optimal *Linear Quadratic Gaussian* (LQG), dengan filter Kalman sebagai estimator yang mampu mengestimasi variabel-variabel yang tidak terukur pada sistem dengan adanya *noise*.

### 1.4 Batasan Masalah

Untuk memfokuskan permasalahan dalam Tugas Akhir ini, maka perlu adanya batasan masalah yang memiliki ruang lingkup sebagai berikut:

1. Plant yang dijadikan objek penelitian adalah unit *boiler-turbine* yang ada di Sydvenska Kraft AB di Malmö, Sweden dengan daya *output* maksimal adalah sebesar 160 MW.
2. Variabel yang dikendalikan adalah tekanan dan ketinggian air.
3. Model matematika yang digunakan diperoleh dari referensi *paper*.
4. Kontroler yang digunakan adalah kontrol optimal *Linear Quadratic Gaussian*.
5. Metode yang digunakan untuk menentukan matriks pembobot  $Q$  dan  $R$  adalah metode *trial and error*.
6. Sistem disimulasikan secara *offline* dengan menggunakan perangkat lunak MATLAB.
7. Perancangan HMI menggunakan *software* Wonderware Intouch.

### 1.5 Metodologi

Pelaksanaan Tugas Akhir ini terbagi atas beberapa tahap dengan rincian sebagai berikut:

1. Studi Literatur  
Sebagai landasan dalam melakukan sebuah penulisan, maka diperlukan teori penunjang yang memadai, baik mengenai ilmu dasar, metode penelitian, teknik analisis, maupun teknik penulisan. Teori penunjang ini bisa diperoleh dari buku referensi, *paper*, jurnal ilmiah baik nasional maupun internasional, serta media *online*. Teori ditekankan pada perancangan sistem *plant* yaitu

*boiler-turbine* serta perancangan kontroler yaitu *Linear Quadratic Gaussian* (LQG). Tahap ini dapat dilakukan di mana saja dan dilakukan kapan saja sepanjang proses pengerjaan Tugas Akhir ini hingga menjelang penulisan buku.

2. Permodelan Sistem

Pada tahap ini, dicari perumusan model matematika linier dari *plant boiler-turbine* dengan cara melakukan linierisasi pada model *nonlinear* yang didapatkan dari referensi *paper*. Permodelan merupakan salah satu tahap yang paling penting dan membutuhkan waktu yang cukup lama dalam proses pengerjaan Tugas Akhir ini. Jika permodelan matematika yang didapatkan kurang tepat, maka analisis sistem yang dihasilkan pun akan menjadi kurang tepat, sehingga bisa menyebabkan kesalahan dalam perancangan kontroler.

3. Perancangan *Simulink Plant*

Setelah memperoleh model matematika yang sesuai maka langkah selanjutnya adalah merancang model matematika ke dalam *simulink*. Pada tahap ini dibutuhkan ketelitian, apabila terdapat kesalahan akan mengakibatkan respon yang terjadi tidak sesuai dengan respon yang sesungguhnya. Hal ini dapat mempengaruhi pengerjaan pada tahap selanjutnya.

4. Perancangan Kontroler

Kontroler yang digunakan dalam Tugas Akhir ini adalah *Linear Quadratic Gaussian* (LQG). Perancangan kontroler dapat dilaksanakan setelah mendapatkan model linier *plant boiler-turbine*.

5. Simulasi

Pada proses simulasi ini, akan dilakukan analisis terhadap data. Analisis yang dilakukan berupa analisis komparatif terhadap respon sistem ketika diberi sinyal masukan berupa sinyal *step*. Analisis ini juga membandingkan antara hasil simulasi dengan teori yang telah dipelajari pada perkuliahan di kelas. Melalui analisis ini, dapat dirumuskan suatu kesimpulan dan rekomendasi terhadap penelitian yang telah dilakukan.

## 6. Penulisan Buku

Tahap ini merupakan bagian akhir dari pengerjaan Tugas Akhir. Penulisan buku ini berupa laporan ilmiah yang mencakup semua proses pengerjaan Tugas Akhir, mulai dari teori yang digunakan hingga kesimpulan dan rekomendasi yang dihasilkan dari penelitian. Semua tahap dari pengerjaan Tugas Akhir ini didokumentasikan secara sistematis sehingga menjadi sebuah pendukung dari rangkaian penjelasan yang ada. Selain buku, juga dibuat *proceeding* sebagai ringkasan dan materi Tugas Akhir presentasi dalam bentuk *power point*. Setelah buku selesai dibuat, dilakukan mekanisme pengesahan yang meliputi pengajuan tanda tangan, *draft* buku, buku, dan *proceeding* Tugas Akhir.

## 1.6 Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam 5 bab yang masing-masing membahas permasalahan yang berhubungan dengan Tugas Akhir yang telah dibuat dengan sistematika penulisan sebagai berikut:

### BAB I

#### PENDAHULUAN

Pada bab ini akan dibahas mengenai latar belakang, perumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika penulisan.

### BAB II

#### DASAR TEORI

Pada bab ini akan dipaparkan teori-teori yang dijadikan referensi dan penunjang terhadap pemecahan suatu masalah pada Tugas Akhir, di antaranya adalah dasar teori tentang fungsi dan cara kerja *boiler-turbine* serta penjelasan tentang kontroler yang digunakan pada Tugas Akhir ini yaitu *Linear Quadratic Gaussian* (LQG).

### BAB III

#### PERANCANGAN SISTEM

Pada bab ini dijelaskan secara rinci mengenai rancangan sistem dan perancangan kontroler yang digunakan pada Tugas Akhir ini.

## **BAB IV**

### **PENGUJIAN DAN ANALISA DATA**

Bab ini berisi pengujian pada tiap-tiap komponen-komponen pembangun sistem serta analisis data hasil simulasi.

## **BAB V**

### **PENUTUP**

Bab ini berisi tentang kesimpulan dan saran-saran untuk pengembangan Tugas Akhir.

*Halaman ini sengaja dikosongkan*

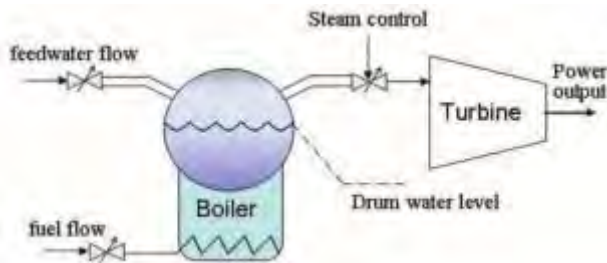
## BAB II

### TINJAUAN PUSTAKA

Bab ini akan membahas mengenai materi dasar dalam penyusunan Tugas Akhir ini. Beberapa hal yang akan dibahas meliputi tinjauan pustaka yang akan membahas mengenai sistem *boiler-turbine*, linearisasi persamaan nonlinear, dan teori kontrol optimal *Linear Quadratic Gaussian*.

#### 2.1 *Boiler-Turbine*

Ketel Uap (*boiler*) adalah alat untuk menghasilkan uap air (*steam*). Uap air yang dihasilkan bukan hanya sekedar uap air saja, tetapi uap air yang bertekanan tinggi. Uap air yang dihasilkan oleh *boiler* tersebut selanjutnya digunakan untuk berbagai macam keperluan, seperti untuk memutar turbin uap, pemanas ruangan, mesin uap, dan lain sebagainya. Jika air dididihkan sampai menjadi *steam*, volumenya akan meningkat sekitar 1.600 kali lipat dan dapat menghasilkan tenaga yang menyerupai bubuk mesiu yang mudah meledak, sehingga *boiler* merupakan peralatan yang harus dikelola dan dijaga dengan sangat baik. Secara proses konversi energi, *boiler* memiliki fungsi untuk mengkonversi energi kimia yang tersimpan di dalam bahan bakar menjadi energi panas. Panas yang diberikan kepada fluida di dalam *boiler* berasal dari proses pembakaran dengan berbagai macam jenis bahan bakar yang dapat digunakan, seperti kayu, batubara, solar/minyak bumi, dan gas. Dengan adanya kemajuan teknologi, energi nuklir pun juga bisa digunakan sebagai sumber panas pada *boiler* [4]. Diagram sederhana dari unit *boiler-turbine* dapat dilihat pada Gambar 2.1.



**Gambar 2.1** Diagram Sederhana dari *Boiler-Turbine Unit*

Dalam sistem *boiler*, air merupakan media utama yang diolah di dalam *boiler* yang selanjutnya akan diproses untuk menghasilkan *steam*. Air di dalam *boiler* dipanaskan oleh panas dari hasil pembakaran bahan bakar (sumber panas lainnya) sehingga terjadi perpindahan panas dari sumber panas tersebut ke air yang mengakibatkan air tersebut menjadi panas atau berubah wujud menjadi uap. Air yang memiliki suhu lebih tinggi memiliki massa jenis yang lebih rendah dibandingkan dengan air yang memiliki suhu lebih rendah, sehingga terjadi perubahan massa jenis air di dalam *boiler*. Air yang memiliki massa jenis yang lebih rendah akan naik, dan sebaliknya air yang memiliki massa jenis yang lebih tinggi akan turun ke dasar. Bejana bertekanan pada *boiler* umumnya menggunakan bahan baja dengan spesifikasi tertentu yang telah ditentukan dalam standard ASME (*The ASME Code Boilers*), terutama untuk penggunaan *boiler* pada industri-industri besar. Dalam sejarah tercatat berbagai macam jenis material digunakan sebagai bahan pembuatan *boiler* seperti tembaga, kuningan, dan besi cor. Namun bahan-bahan tersebut sudah lama ditinggalkan karena alasan ekonomis dan juga ketahanan material yang sudah tidak sesuai dengan kebutuhan industri.

Sistem *boiler* terdiri dari sistem *feedwater* (air umpan), sistem *steam* (uap air), dan sistem *fuel* (bahan bakar). Sistem air umpan menyediakan air secara otomatis sesuai dengan kebutuhan *steam*. Air umpan merupakan air yang disuplai ke *boiler* untuk diubah menjadi *steam*. Ada dua sumber air umpan, yaitu yang pertama adalah kondensat atau *steam* yang mengembun ke proses, dan sumber air umpan yang kedua adalah air *make up* (air baku yang sudah diolah) yang harus diumpangkan dari luar ruang *boiler* ke plant proses. Untuk mendapatkan efisiensi boiler yang lebih tinggi, digunakan *economizer* untuk memanaskan awal air umpan menggunakan limbah panas pada gas buang. Sistem *steam* berfungsi mengumpulkan dan mengontrol produksi *steam* dalam *boiler*. Sistem bahan bakar adalah, semua peralatan yang digunakan untuk menyediakan bahan bakar sehingga *boiler* dapat menghasilkan panas yang dibutuhkan. Peralatan yang digunakan pada sistem bahan bakar tergantung pada jenis bahan bakar yang digunakan. Uap yang dihasilkan boiler mempunyai temperatur dan tekanan tertentu sedemikian rupa sehingga dapat bersifat efisien mungkin untuk digunakan.

## **2.2 Model Matematika Boiler-Turbine**

Model *boiler-turbine* yang akan digunakan pada pengerjaan Tugas Akhir ini merupakan sistem dengan model matematika *nonlinear*. Sistem



*boiler-turbine* memiliki tiga variabel *state* yang telah dikembangkan dan diteliti oleh Bell dan Astrom pada tahun 1987. Model ini mengatur pengoperasian *boiler-turbine* saat mempertahankan penekanan pada struktur yang lebih sederhana dan merupakan perluasan model *nonlinear* berbentuk dua *state* yang mempresentasikan tekanan drum dan dinamika daya. Penyertaan persamaan penguapan ekstra dan dinamika fluida dapat menyampaikan gambaran tentang dinamika level drum air.

Parameter-parameter perhitungan yang akan digunakan pada *boiler-turbine* ini berasal dari data dinamis yang diukur dari plant Pembangkit Listrik Tenaga Uap (PLTU) yang berada di Sydvenska Kraft AB daerah dekat Malmo, Swedia. Pembangkit Listrik Tenaga Uap (PLTU) ini berbahan bakar minyak bumi dan dapat menghasilkan daya output maksimal sebesar 160 MW. Meskipun model *boiler-turbine* ini termasuk sistem orde rendah, model ini mampu menangkap dinamika perilaku utama yang sesuai dengan *real plant*.

*State* sistem pada *boiler-turbine* didefinisikan sebagai vektor

kolom  $x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$ , di mana:

$x_1$  = tekanan drum (Kg/cm<sup>2</sup>),

$x_2$  = daya *output* (MW),

$x_3$  = masa jenis fluida (Kg/m<sup>3</sup>)

Model matematika *nonlinear* dalam bentuk persamaan *state space* dapat dituliskan sebagai berikut [3]:

$$\begin{aligned} \dot{x}_1 &= -0,0018u_2x_1^{\frac{9}{8}} + 0,9u_1 - 0,15u_3 \\ \dot{x}_2 &= (0,073u_2 - 0,016)x_1^{\frac{9}{8}} - 0,1x_2 \\ \dot{x}_3 &= (141u_3 - (1,1u_2 - 0,19)x_1)/85 \\ y_1 &= x_1 \\ y_2 &= x_2 \\ y_3 &= 0,05(0,23073x_3 + 100\alpha_{cs} + q_e/9 - 67,975) \end{aligned} \quad (2.1)$$

dengan:

$$\alpha_{cs} = \frac{(1-0,001538x_3)(0,8x_1-25,6)}{x_3(1,0394-0,0012304x_1)}$$

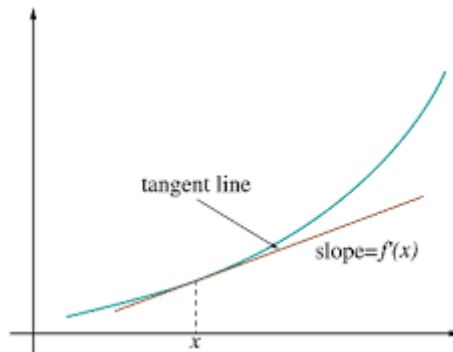
$$q_e = (0,854u_2 - 0,147)x_1 + 45,59u_1 - 2,514u_3 - 2,096$$

Untuk  $u_1$ ,  $u_2$ , dan  $u_3$  adalah *input* sistem yang merupakan posisi *control valve* yang terdiri dari *control valve* untuk aliran bahan bakar, *control valve* untuk kontrol uap, dan *control valve* untuk aliran *feedwater*. *Output* sistem  $y_1$ ,  $y_2$ , dan  $y_3$  berturut-urut adalah tekanan drum (Kg/cm<sup>2</sup>), daya *output* (MW), dan ketinggian fluida (m). Sedangkan  $\alpha_{cs}$  merupakan kualitas uap dan  $q_e$  sebagai laju penguapan (Kg/s).

### 2.3 Linierisasi

Sebuah sistem *nonlinear* dapat dicari solusi penyelesaiannya melalui pendekatan model linier. Dalam hal ini, persamaan-persamaan (model) *nonlinear* pada sistem terlebih dahulu dilinierkan pada titik kerja yang telah ditentukan.

Dari hasil linierisasi model, dihasilkan suatu matriks konstanta yang sama dengan suatu sistem kontrol linier untuk setiap daerah kerja, sehingga mempermudah pengontrolan. Dengan linierisasi, akan diperoleh perilaku lokal dari sistem pada daerah kerja nominal dengan menggunakan matriks Jacobi.



**Gambar 2.2** Linierisasi  $y=f(x)$  di Sekitar  $x^*$

Sebuah sistem yang memiliki model matematika nonlinier bisa dinyatakan dengan persamaan di bawah ini:

$$\dot{x} = f(x) \quad (2.2)$$

dengan  $y = f(x)$  merupakan kurva yang ditunjukkan pada Gambar 2.2. Pada titik  $x^*$ , garis singgung merepresentasikan pendekatan linier  $f(x)$  di sekitar  $x^*$ . Hasil linierisasi dari sistem dapat dituliskan sebagai berikut:

$$(f(x) - f(x^*)) = f'(x - x^*) \quad (2.3)$$

$$\delta y = f(\delta y) = \left. \frac{df(x)}{dx} \right|_{x=x^*} \delta x$$

Titik kerja  $x^*$  disebut sebagai titik pusat dari hasil linierisasi. Jika titik kerja sistem berada pada titik *origin*, dan sistem dilinierisasi pada titik kerja yang telah ditentukan, maka  $f(x^*) = 0$ , sehingga Persamaan (2.3) menjadi:

$$f(x) = \dot{x} = \left. \frac{df(x)}{dx} \right|_{x=x^*} x \quad (2.4)$$

Jika pada sistem nonlinier lup terbuka terdapat *state* berjumlah  $n$ , maka linierisasi sistem pada titik kerja  $(x_1^*, x_2^*, \dots, x_n^*)$  menjadi sebagai berikut:

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(x_1, x_2, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_1(x_1, x_2, \dots, x_n)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x_1, x_2, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_n(x_1, x_2, \dots, x_n)}{\partial x_n} \end{bmatrix} \bigg|_{\substack{x_1=x_1^* \\ \vdots \\ x_n=x_n^*}} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

atau dapat juga ditulis dalam bentuk umum sebagai berikut:

$$\dot{x} = Ax \quad (2.5)$$

dengan  $x$  adalah *state* sistem dalam bentuk vector kolom  $nx1$ , dan  $A$  adalah matriks  $nxn$ , dengan  $n$  adalah jumlah *state*, atau dapat dinyatakan sebagai berikut:

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

$$A = \begin{bmatrix} \frac{\partial f_1(x_1, x_2, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_1(x_1, x_2, \dots, x_n)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x_1, x_2, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_n(x_1, x_2, \dots, x_n)}{\partial x_n} \end{bmatrix}$$

Dengan demikian, maka sistem dengan model matematika nonlinier dapat didekati menjadi sistem linier berdasarkan hasil linierisasi pada titik kerja yang telah ditentukan, dengan matriks **A** sebagai matriks Jacobi dari  $f(x)$ , dan dapat direpresentasikan dalam persamaan (2.6), yaitu sebagai berikut:

$$\dot{x} = f(x) + h(x, u) \quad (2.6)$$

Jika jumlah *state* dalam sistem adalah  $n$ , maka linierisasi sistem pada titik kerja  $(x_1^*, x_2^*, \dots, x_n^*, u_1^*, u_2^*, \dots, u_n^*)$  adalah sebagai berikut:

$$\dot{x} = Ax + Bu \quad (2.7)$$

dengan elemen dari vector kolom **x**, matriks **A**, dan matriks **B** adalah sebagai berikut:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}$$

$$A = \left[ \begin{array}{ccc} \frac{\partial f_1(x)}{\partial x_1} & \dots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x)}{\partial x_1} & \dots & \frac{\partial f_n(x)}{\partial x_n} \end{array} \right] \bigg|_{x_1=x_1^*}$$

$$\mathbf{B} = \left\| \begin{bmatrix} \frac{\partial h_1(x, u)}{\partial u} \\ \vdots \\ \frac{\partial h_n(x, u)}{\partial u} \end{bmatrix} \right\|_{\substack{x=x^* \\ u=u^*}}$$

dan persamaan output sistem adalah

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}u \quad (2.8)$$

dengan elemen vektor kolom dari  $\mathbf{y}$ , matriks  $\mathbf{C}$ , dan matriks  $\mathbf{D}$  adalah sebagai berikut:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

$$\mathbf{C} = \left\| \begin{bmatrix} \frac{\partial g_1(x)}{\partial x_1} & \dots & \frac{\partial g_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_n(x)}{\partial x_1} & \dots & \frac{\partial g_n(x)}{\partial x_n} \end{bmatrix} \right\|_{x_1=x_1^*}$$

$$\mathbf{D} = \left\| \begin{bmatrix} \frac{\partial i_1(x, u)}{\partial u} \\ \vdots \\ \frac{\partial i_n(x, u)}{\partial u} \end{bmatrix} \right\|_{\substack{x=x^* \\ u=u^*}}$$

## 2.4 Sistem Kontrol Optimal

Kontrol optimal berhubungan dengan permasalahan dalam menentukan hukum kontrol untuk suatu sistem sehingga kriteria keoptimalan tertentu dapat terpenuhi. Permasalahan kontrol optimal melibatkan fungsi biaya yang merupakan fungsional atas *state* dan variabel kontrol. Kontrol optimal adalah himpunan dari persamaan

diferensial yang merupakan lintasan dari variabel kontrol yang meminimalkan fungsi biaya.

Kontrol optimal merupakan perluasan dari kalkulus variasi, yaitu metode optimasi matematika untuk menurunkan kebijakan pengendalian. Metode ini sebagian besar dikembangkan oleh Lev Pontryagin dan rekan-rekannya di Uni Soviet, serta Richard Bellman di Amerika Serikat.

Kontrol optimal berkaitan dengan masalah bagaimana menyelesaikan hukum kontrol untuk sistem tertentu seperti bahwa kriteria optimalitas tertentu dicapai. Sebuah permasalahan kontrol mencakup biaya fungsional yang merupakan fungsi dari variabel. Sebuah kontrol optimal adalah satu set persamaan diferensial yang menggambarkan jalan dari variabel-variabel kontrol yang meminimalkan biaya fungsional. Kontrol optimal dapat diturunkan dengan menggunakan prinsip maksimum Pontryagin (suatu kondisi yang diperlukan), atau dengan menyelesaikan persamaan Hamilton.

Bentuk umum dari permasalahan kontrol optimal diberikan sebagai berikut. Sistem dinamik *nonlinear* dideskripsikan dalam bentuk persamaan differensial:

$$\dot{x}(t) = f(x(t), u(t), t) \quad (2.5)$$

dengan *state*  $x(t) \in \mathbf{R}^n$ , *control input*  $u(t) \in \mathbf{R}^m$ , dan indeks performansi:

$$J(t) = \phi(x(T), T) + \int_{t_0}^T L(x(t), u(t), t) dt \quad (2.6)$$

di mana:

$t_0$  adalah waktu awal (tetap),

$T$  adalah waktu akhir (bebas),

$L(x(t), u(t), t)$  adalah fungsi biaya pada selang waktu antara  $[t_0, T]$ ,

$\phi(x(T), T)$  adalah fungsi biaya pada waktu akhir yang bergantung hanya pada  $T$  dan  $x(t)$ .

Permasalahan kontrol optimal adalah untuk mencari input  $u^*(t)$  pada selang waktu  $[t_0, T]$  yang membawa persamaan (2.5) sepanjang lintasan  $x^*(t)$  sehingga nilai dari indeks performansi (2.6) menjadi minimum, dan

$$\psi(x(T), T) = 0 \quad (2.7)$$

di mana  $\psi(x(T), T)$  merupakan fungsi pembatas pada *state* akhir dengan  $\psi \in \mathbf{R}^p$  diberikan [5].

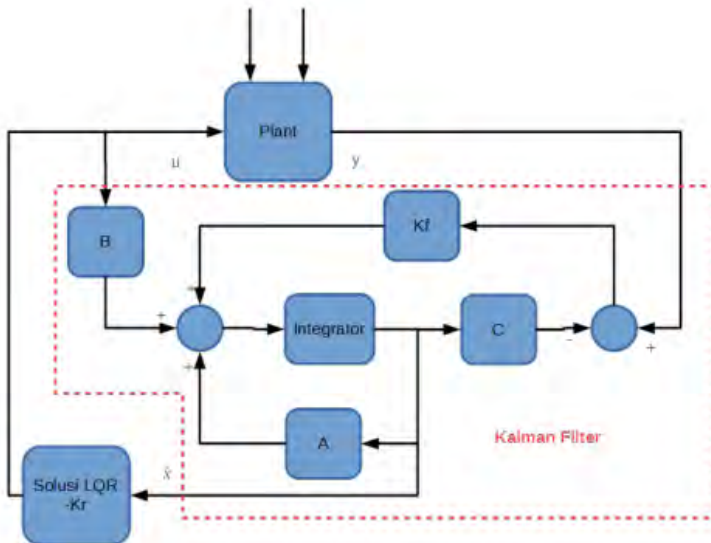
#### 2.4.1 Kontrol Optimal *Linear Quadratic Gaussian*

Masalah desain kontrol dalam dunia riil seringkali kita tidak dapat mengukur seluruh variabel *state* dari plant yang diberikan. Maka dari itu, LQR, meskipun mempunyai margin penguatan dan margin fasa yang sangat bagus, namun sulit direalisasikan karena menggunakan semua variabel *state* sebagai umpan balik (*feedback*),  $u = -Kx$ . Untuk situasi praktis, hanya informasi parsial dari *state* pada plant yang diberikan dapat diakses/diukur untuk umpan balik. Untuk mengestimasi variabel *state* yang tak terukur, maka digunakan *Linear Quadratic Estimator* atau filter Kalman. Untuk mengganti *state* pada sinyal kontrol optimal  $u = -Kx$ , maka digunakan *state* hasil estimasi dari filter Kalman untuk mendesain sinyal kontrol yang optimal, yaitu dengan *Linear Quadratic Gaussian*.

Kontrol optimal *Linear Quadratic Gaussian* (LQG) adalah teknik kontrol modern yang diimplementasikan dalam bentuk ruang dan waktu (*state space*). Pada awalnya *state space* digunakan untuk mendesain regulator optimal atau LQR. Seiring berjalannya waktu, kontrol LQR mengalami pengembangan yaitu dengan mengikutsertakan estimator optimal (filter Kalman) yang berfungsi untuk mengestimasi variabel keadaan dan menyaring *noise*. *Noise* yang mempengaruhi kontrol LQG berupa *white noise* yang mempunyai spektrum frekuensi sama untuk berbagai frekuensi.

Pengaruh *noise* pada kontroler menyebabkan tidak optimalnya sinyal kontrol pada aktuator, maka untuk mengoptimalkan input pada proses ( $u$ ) diperlukan penguat kontroler regulator ( $K$ ) dan penguat estimator kalman filter ( $L$ ). Pencarian harga regulator dilakukan dengan cara seolah-olah sistem bersifat deterministik yaitu dengan metode LQR, sedangkan pencarian harga optimal filter Kalman dilakukan dengan memperlakukan sistem bersifat stokastik, yaitu dengan mendapatkan index performansi kesalahan minimum yang terkecil, atau dituliskan:

$$J(t) = \phi(x(T), T) + \int_{t_0}^T L(x(t), u(t), t) dt$$



**Gambar 2.3** Diagram Blok LQG

Tujuan pengendalian dengan menggunakan kontroler LQG yaitu untuk mendesain sinyal kontrol  $u$  dengan hanya berasal dari informasi yang dapat diukur sehingga ketika dimasukkan ke plant sebagai input, sistem keseluruhan akan stabil dan indeks performansi akan minimum.

Estimator merupakan fungsi transfer estimasi dari proses plant yang akan dikendalikan dengan menyertakan filter kalman sebagai estimator variabel keadaan dan penyaring *noise*. Tujuan estimator adalah sebagai plant acuan untuk mengestimasi keadaan dengan sifat *linear time invariant*, artinya nilai parameter matriks A, B, C, dan D adalah tetap pada semua waktu. Kelebihan estimator adalah dapat mengendalikan *multivariable input* dan *output* pada plant yang saling berkaitan dalam satu kendali. Besarnya harga penguat regulator adalah:

$$K = R^{-1}B^T P$$

Nilai P harus definit positif dan untuk mencari nilai P maka digunakan solusi persamaan *Riccati*:

$$0 = A^T P + P A + Q - P B R^{-1} B^T P$$



### 2.4.2 Controllability

*Complete state controllability* menggambarkan kemampuan dari input eksternal untuk membawa setiap *state* dari sistem dari keadaan awal ke keadaan akhir pada selang waktu tertentu. Sebuah sistem dapat dikatakan *controllable* jika dan hanya jika semua *state* dari sistem dapat berubah dengan mengubah masukan sistem.

Sebuah *state*  $x_0$  adalah *controllable* pada waktu  $t_0$  jika untuk selang waktu yang terbatas  $t_1$  terdapat input  $u(t)$  yang membawa *state*  $x(t)$  dari  $x_0$  ke titik asal atau titik origin pada waktu  $t_1$ .

Untuk sistem LTI (*Linear Time Invariant*), sistem dapat dikatakan *controllable* jika matriks *controllability*,  $W$ , memiliki *rank* sama dengan dimensi dari matriks  $A$ . dengan matriks  $W$  adalah [6]

$$W = [B \quad AB \quad \dots \quad A^{n-1}B]$$

### 2.4.3 Observability

Sebuah sistem dengan keadaan awal  $x(t_0)$  adalah *observable* jika dan hanya jika nilai dari *initial state* dapat ditentukan dari output sistem  $y(t)$  yang telah teramati/*observed* pada interval waktu  $t_0 < t < t_f$ . Jika *initial state* tidak dapat ditentukan, maka sistem tersebut adalah *unobservable*.

Sebuah sistem dikatakan *completely observable* jika semua *initial state* dari sistem dapat diamati/*observable*. Sistem yang tidak memenuhi kriteria tersebut adalah sistem yang *unobservable*.

Untuk melihat apakah sistem *observable* dapat dilihat dari matriks  $A$  dan matriks  $C$ . Sistem *observable* jika *rank* dari matriks *observability* adalah sama dengan dimensi dari matriks  $A$ . Matriks *observability* adalah matriks dengan persamaan sebagai berikut [6]:

$$M = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

## 2.5 Wonderware Intouch

Wonderware adalah merek otomasi industri dan produk informasi perangkat lunak, yang dimiliki oleh Invensys Manajemen Operasi (IOM), yaitu sebuah divisi dari perusahaan Inggris teknologi Invensys. Lebih dari

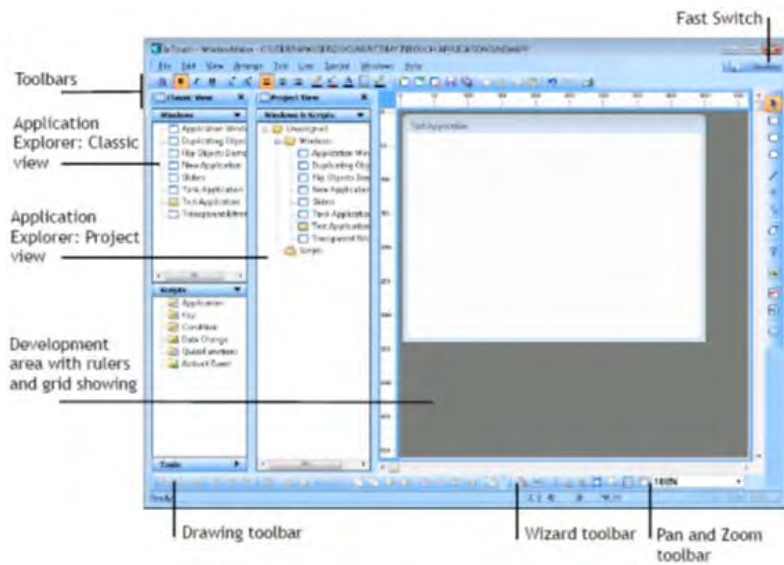
500.000 lisensi perangkat lunak telah terjual dan diinstal di lebih dari 100.000 tanaman di seluruh dunia. Wonderware perangkat lunak yang digunakan di banyak industri – termasuk utilitas, pertambangan, minyak dan gas, makanan dan minuman, farmasi, transportasi, pulp dan kertas, semikonduktor dan logam.

Berbasis di Lake Forest, California, Wonderware mengoperasikan jaringan global berwenang, 160 kantor distributor melayani 130 negara. Lokal dan regional integrasi sistem dan layanan implementasi disediakan melalui jaringan 3500 lebih independen sistem perusahaan integrator.

Wonderware InTouch memperkenalkan perangkat lunak pada tahun 1987. Hal ini diklaim menjadi yang pertama antarmuka mesin manusia (HMI) berdasarkan sistem operasi Microsoft Windows.

Pada dasarnya InTouch adalah *software Human Machine Interface* yg dilengkapi dg fitur dasar SCADA *software*. Untuk menggunakan wonderware intouch ada 3 komponen penyusun utama yg harus diketahui yaitu:

- *Intouch application manager*, berfungsi untuk mengorganisasikan aplikasi yang akan dibuat.
- *Intouch window maker*, suatu *development environment* dari intouch, kita dapat membuat halaman-halaman HMI dengan grafik yang *object-oriented* untuk menciptakan layar tampilan yang dapat bergerak dan dapat menerima masukan dari pengguna.
- *Intouch window viewer* suatu *runtime environment* yang dapat menampilkan layar grafik yang telah dibuat pada *windowmaker*.



**Gambar 2.4** Tampilan *Window Maker* Pada *Software* Wonderware Intouch

*Halaman ini sengaja dikosongkan*

## BAB III

### PERANCANGAN SISTEM

Pada bab ini akan dibahas linierisasi model matematika *boiler-turbine* pada titik kerja yang telah ditentukan. Setelah dilakukan linierisasi pada titik kerja yang telah ditentukan sebelumnya, maka didapatkan model matematika linier *boiler-turbine* dalam bentuk *state space*. Model matematika linier dalam bentuk *state space* yang sudah didapatkan akan digunakan untuk merancang sistem kontrol optimal *Linear Quadratic Gaussian*. Perancangan sistem dilakukan dengan menggunakan perangkat lunak MATLAB untuk simulasi dan Wonderware InTouch untuk merancang *Human Machine Interface* (HMI).

#### 3.1 Linierisasi Model Matematika *Boiler-Turbine*

Model matematika *nonlinear boiler-turbine* dinamik pada Persamaan (2.1), sistem akan dilinerisasi pada titik kerja yang telah ditentukan. Model linier *boiler-turbine* pada titik kerja  $x_1^*$ ,  $x_2^*$ ,  $x_3^*$  dan  $u_1^*$ ,  $u_2^*$ ,  $u_3^*$  adalah

$$\dot{x} = Ax + Bu \quad (3.1)$$

dengan

$$A = \left[ \begin{array}{ccc} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \frac{\partial f_1(x)}{\partial x_3} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \frac{\partial f_2(x)}{\partial x_3} \\ \frac{\partial f_3(x)}{\partial x_1} & \frac{\partial f_3(x)}{\partial x_2} & \frac{\partial f_3(x)}{\partial x_3} \end{array} \right]_{x=x^*}$$

$$B = \left[ \begin{array}{ccc} \frac{\partial h_1(x)}{\partial u_1} & \frac{\partial h_1(x)}{\partial u_2} & \frac{\partial h_1(x)}{\partial u_3} \\ \frac{\partial h_2(x)}{\partial u_1} & \frac{\partial h_2(x)}{\partial u_2} & \frac{\partial h_2(x)}{\partial u_3} \\ \frac{\partial h_3(x)}{\partial u_1} & \frac{\partial h_3(x)}{\partial u_2} & \frac{\partial h_3(x)}{\partial u_3} \end{array} \right]_{u=u^*}$$

dan persamaan output sistem adalah

$$y = \mathbf{C}x + \mathbf{D}u \quad (3.2)$$

dengan

$$\mathbf{C} = \left\| \begin{array}{ccc} \frac{\partial g_1(x)}{\partial x_1} & \frac{\partial g_1(x)}{\partial x_2} & \frac{\partial g_1(x)}{\partial x_3} \\ \frac{\partial g_2(x)}{\partial x_1} & \frac{\partial g_2(x)}{\partial x_2} & \frac{\partial g_2(x)}{\partial x_3} \\ \frac{\partial g_3(x)}{\partial x_1} & \frac{\partial g_3(x)}{\partial x_2} & \frac{\partial g_3(x)}{\partial x_3} \end{array} \right\|_{x=x^*}$$

$$\mathbf{D} = \left\| \begin{array}{ccc} \frac{\partial i_1(x)}{\partial u_1} & \frac{\partial i_1(x)}{\partial u_2} & \frac{\partial i_1(x)}{\partial u_3} \\ \frac{\partial i_2(x)}{\partial u_1} & \frac{\partial i_2(x)}{\partial u_2} & \frac{\partial i_2(x)}{\partial u_3} \\ \frac{\partial i_3(x)}{\partial u_1} & \frac{\partial i_3(x)}{\partial u_2} & \frac{\partial i_3(x)}{\partial u_3} \end{array} \right\|_{u=u^*}$$

Dilakukan linierisasi pada titik – titik kerja yang telah ditentukan, yaitu pada titik  $x = \begin{bmatrix} 108 \\ 66,65 \\ 428 \end{bmatrix}$ ,  $u = \begin{bmatrix} 0,34 \\ 0,69 \\ 0,433 \end{bmatrix}$ , dan  $y = \begin{bmatrix} 108 \\ 66,65 \\ 0 \end{bmatrix}$ , sehingga didapatkan elemen matriks  $\mathbf{A}$ :

$$\frac{\partial f_1(x)}{\partial x_1} = -0,0018u_2x_1^{1/8} \quad \frac{\partial f_1(x)}{\partial x_2} = 0 \quad \frac{\partial f_1(x)}{\partial x_3} = 0$$

$$\frac{\partial f_2(x)}{\partial x_1} = (0,08u_2 - 0,018)x_1^{1/8} \quad \frac{\partial f_2(x)}{\partial x_2} = 0 \quad \frac{\partial f_2(x)}{\partial x_3} = 0$$

$$\frac{\partial f_3(x)}{\partial x_1} = \frac{-(1,1u_2-0,19)}{85} \quad \frac{\partial f_3(x)}{\partial x_2} = 0 \quad \frac{\partial f_3(x)}{\partial x_3} = 0$$

Elemen matriks  $\mathbf{B}$ :

$$\frac{\partial h_1(x,u)}{\partial u_1} = 0,9 \quad \frac{\partial h_1(x,u)}{\partial u_2} = -0,0018x_1^{9/8} \quad \frac{\partial h_1(x,u)}{\partial u_3} = -0,15$$

$$\begin{aligned}\frac{\partial h_2(x,u)}{\partial u_1} &= 0 & \frac{\partial h_2(x,u)}{\partial u_2} &= 0,073x_1^{9/8} & \frac{\partial h_2(x,u)}{\partial u_3} &= 0 \\ \frac{\partial h_3(x,u)}{\partial u_1} &= 0 & \frac{\partial h_3(x,u)}{\partial u_2} &= \frac{-1,1x_1}{85} & \frac{\partial h_3(x,u)}{\partial u_3} &= \frac{141}{85}\end{aligned}$$

Elemen matriks **C**:

$$\begin{aligned}\frac{\partial g_1(x,u)}{\partial x_1} &= 1 & \frac{\partial g_1(x,u)}{\partial x_2} &= 0 & \frac{\partial g_1(x,u)}{\partial x_3} &= 0 \\ \frac{\partial g_2(x,u)}{\partial x_1} &= 0 & \frac{\partial g_2(x,u)}{\partial x_2} &= 1 & \frac{\partial g_2(x,u)}{\partial x_3} &= 0 \\ \frac{\partial g_3(x,u)}{\partial x_1} &= \frac{\left(\frac{0,05(0,08u_2-0,147)}{9}\right)+5((0,8-0,0012x_3)b-cd)}{(1,0394x_3-(0,0012x_1x_3))^2} \\ \frac{\partial g_3(x,u)}{\partial x_2} &= 0 \\ \frac{\partial g_3(x,u)}{\partial x_3} &= \frac{0,05*0,13073+5eb-cf}{(1,039x_3-0,0012x_1x_3)^2}\end{aligned}$$

dengan

$$b = 1,0394x_3 - 0,0012x_1x_3$$

$$c = 0,8x_1 - 0,0012x_1x_3 - 25,6 + 0,0393x_3$$

$$d = -0,0012x_3$$

$$f = 0,0393 - 0,0012x_1$$

$$g = 1,0393 - 0,0012x_1$$

Elemen matriks **D**:

$$\begin{aligned}\frac{\partial i_1(x,u)}{\partial u_1} &= 0 & \frac{\partial i_1(x,u)}{\partial u_2} &= 0 & \frac{\partial i_1(x,u)}{\partial u_3} &= 0\end{aligned}$$

$$\frac{\partial i_2(x,u)}{\partial u_1} = 0$$

$$\frac{\partial i_2(x,u)}{\partial u_2} = 0$$

$$\frac{\partial i_2(x,u)}{\partial u_3} = 0$$

$$\frac{\partial i_3(x,u)}{\partial u_1} = 0,2533$$

$$\frac{\partial i_3(x,u)}{\partial u_2} = 0,0047x_1$$

$$\frac{\partial i_3(x,u)}{\partial u_3} = -0,014$$

Dari hasil linierisasi pada titik-titik kerja yang telah ditentukan tersebut, selanjutnya disubstitusikan nilai  $x$  dan  $u$  pada titik kerja yang telah ditentukan, sehingga didapatkan matriks

$$\dot{x} = \mathbf{A}x + \mathbf{B}u \quad (3.3)$$

dengan

$$\mathbf{A} = \begin{bmatrix} -0,0025 & 0 & 0 \\ 0,0694 & -0,1 & 0 \\ -0,0067 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0,9 & -0,349 & -0,15 \\ 0 & 14,16 & 0 \\ 0 & -1,398 & 1,659 \end{bmatrix}$$

dan persamaan output sistem adalah

$$y = \mathbf{C}x + \mathbf{D}u \quad (3.4)$$

dengan

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0,0063 & 0 & 0,0047 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0,2533 & 0,5124 & -0,014 \end{bmatrix}$$

Persamaan (3.3) dan persamaan (3.4) adalah persamaan linier dari *boiler-turbine* dalam bentuk persamaan *state space*.



### 3.2 Pengujian *Controllability* dan *Observability*

Matriks *controllability* bergantung pada matriks **A** dan **B** sedangkan matriks *observability* bergantung pada matriks **A** dan **C**.

Sebuah sistem dapat dikatakan *fully controllable* jika *rank* dari matriks *controllability* adalah sama dengan jumlah *state* dari sistem. Matriks *controllability* dapat dicari dari persamaan [6]

$$W = [B \quad AB \quad \dots \quad A^{n-1}B]$$

Dengan demikian, nilai dari matriks **W** adalah:

$$W = [B \quad AB \quad A^2B]$$

$$W = \begin{bmatrix} 0,9 & -0,34 & -0,15 & -0,002 & 0,0009 & 0,0004 & 0 & 0 & 0 \\ 0 & 14,16 & 0 & 0,0625 & -1,44 & -0,01 & -0,006 & 0,14 & 0,001 \\ 0 & -1,39 & 1,65 & -0,006 & 0,002 & 0,001 & 0 & 0 & 0 \end{bmatrix}$$

*Rank* dari matriks **W** adalah 3, yaitu sama dengan jumlah *state* nya, sehingga sistem dapat dikatakan *controllable*.

Sebuah sistem dapat dikatakan *fully observable* jika *rank* dari matriks *observability* adalah sama dengan jumlah *state* dari sistem. Matriks *observability* dapat dicari dari persamaan [6]

$$M = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Dengan demikian, nilai dari matriks **M** adalah

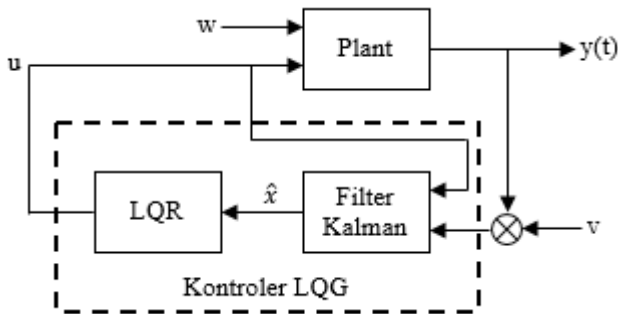
$$M = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix}$$

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0,006 & 0 & 0,004 \\ -0,002 & 0 & 0 \\ 0,069 & -0,1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -0,007 & 0,01 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

*Rank* dari matriks  $M$  adalah 3, yaitu sama dengan jumlah *state* nya, sehingga sistem dapat dikatakan *controllable*.

### 3.3 Perancangan Kontroler

*Linear Quadratic Gaussian* merupakan salah satu teknik kontrol modern yang menggunakan model plant dalam bentuk *state space*. Sistem dengan model *state space* yang dipengaruhi oleh *noise*, yaitu *process noise*  $w$  dan *measurement noise*  $v$ , seperti pada persamaan berikut: [7]



**Gambar 3.1** Diagram Blok LQG Regulator

$$\dot{x} = Ax + Bu + Gw \quad (3.3)$$

$$y = Cx + Du + v \quad (3.4)$$

Dari persamaan (3.3) dan persamaan (3.4),  $w$  adalah gangguan di sinyal kontrol atau *process noise* (*disturbances*), sedangkan  $v$  adalah *measurement noise*. Biasanya  $v$  dan  $w$  seringkali diasumsikan tidak

berkorelasi dan masing masing mempunyai *zero mean Gaussian stochastic* dengan *power spectral density* bernilai konstan. Nilai  $v$  dan  $w$  disebut juga sebagai *white noise* dengan *covariances*

$$E[v(t).v^T(t)] = V\delta(t - \tau) \quad (3.5)$$

$$E[w(t).w^T(t)] = W\delta(t - \tau) \quad (3.6)$$

Dari persamaan (3.5) dan persamaan (3.6) di atas, menunjukkan bahwa asumsi  $w$  dan  $v$  tidak saling berkorelasi.

$$E[v(t).w^T(t)] = 0$$

$$E[w(t).v^T(t)] = 0$$

LQG regulator terdiri dari *kalman state estimator* (filter Kalman) dan *optimal feedback gain* (LQR). *Rank* matriks *controllability* dan matriks *observability* dari sistem harus *full rank*, sehingga sistem *fully state controllable* dan *fully state observable*. *State estimator* digunakan untuk mengestimasi semua *state* sistem.

Performansi dari kontrol LQG dapat ditentukan dari indeks performansi sebagai berikut: [7]

$$J(u) = \int_0^\infty (x^T Q x + u^T R u) dt \quad (3.7)$$

Nilai  $J(u)$  yang minimum merepresentasikan energi yang minimum dari sinyal kontrol  $u$  dan perubahan yang minimum dari variabel *state*.  $Q$  dan  $R$  adalah matriks bobot yang pemilihannya disesuaikan dengan syarat batas sistem. Matriks  $Q$  dan  $R$  dapat dimasukkan ke dalam persamaan Riccati [7]:

$$A^T P + P A - P B R^{-1} B^T P + Q = 0 \quad (3.8)$$

Solusi dari persamaan Riccati adalah matriks  $P$ . Matriks  $P$  adalah matriks dengan nilai definit positif yang digunakan untuk mencari matriks umpan balik optimal  $K$  [7]:

$$K = R^{-1} B^T P \quad (3.9)$$

Matriks digunakan untuk mencari nilai sinyal kontrol  $u$  yang paling optimal [7]

$$u = -Kx \quad (3.10)$$

Untuk menentukan nilai matriks pembobot  $Q$  dan  $R$  dapat dilakukan dengan cara coba-coba atau *trial and error*. Dasar acuan dari pemilihan matriks  $Q$  dan  $R$  ini adalah sebagai berikut:

- ✓ Matriks  $Q$  adalah matriks simetris, semidefinit positif dan real ( $Q \geq 0$ ). Dipilih model matriks diagonal untuk menyederhanakan penyelesaian. Nilai dari variabel kunci diperoleh dengan cara mencoba (trial and error), sehinggadiperoleh solusi yang terbaik.
- ✓ Matriks  $R$  adalah matriks simetris, definit positif dan real ( $R \geq 0$ ). Dipilih matriks berbentuk diagonal, sehingga dapat menyederhanakan perhitungan dan menjaga sistem agar selalu kokoh atau *robust*.

Setelah dilakukan metode *trial and error* untuk mendapatkan matriks  $Q$  dan  $R$ , maka dipilih satu nilai matriks  $Q$  dan  $R$  yang dianggap paling baik. Nilai  $Q$  dan  $R$  yang dipilih adalah:

$$Q = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,01 \end{bmatrix}$$

$$R = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{bmatrix}$$

Nilai matriks  $Q$  dan  $R$  yang sudah dipilih selanjutnya disubtitusikan ke dalam persamaan (3.8) untuk mendapatkan solusi dari penyelesaian persamaan Riccati, yaitu matriks  $P$  [7].

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

$$\begin{aligned}
& \begin{bmatrix} -0,0025 & 0,0694 & -0,0067 \\ 0 & -0,1 & 0 \\ 0 & 0 & 0 \end{bmatrix} P + P \begin{bmatrix} -0,0025 & 0 & 0 \\ 0,0694 & -0,1 & 0 \\ -0,0067 & 0 & 0 \end{bmatrix} \\
& - P \begin{bmatrix} 0,9 & -0,349 & -0,15 \\ 0 & 14,16 & 0 \\ 0 & -1,398 & 1,659 \end{bmatrix} \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,01 \end{bmatrix} \begin{bmatrix} 0,9 & 0 & 0 \\ -0,349 & 14,16 & -1,398 \\ -0,15 & 0 & 1,659 \end{bmatrix} P \\
& + \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,01 \end{bmatrix} = 0
\end{aligned}$$

Dengan menggunakan bantuan *software* Matlab maka didapatkan solusi dari persamaan Riccati, yaitu matriks  $P$  sebagai berikut:

$$P = \begin{bmatrix} 1,0009 & 0,0224 & -0,0316 \\ 0,0224 & 0,0376 & 0,0199 \\ -0,0316 & 0,0199 & 0,5341 \end{bmatrix}$$

Selanjutnya matriks  $P$  yang sudah didapatkan disubstitusikan ke dalam persamaan (3.9) untuk mendapatkan nilai matriks penguat umpan balik optimal, yaitu  $K$ .

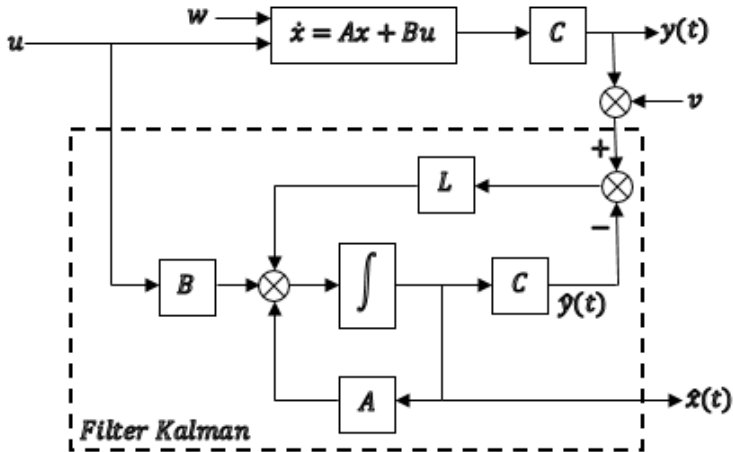
$$K = R^{-1}B^T P$$

$$\begin{aligned}
K &= \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,01 \end{bmatrix} \begin{bmatrix} 0,9 & 0 & 0 \\ -0,349 & 14,16 & -1,398 \\ -0,15 & 0 & 1,659 \end{bmatrix} \\
&= \begin{bmatrix} 1,0009 & 0,0224 & -0,0316 \\ 0,0224 & 0,0376 & 0,0199 \\ -0,0316 & 0,0199 & 0,5341 \end{bmatrix} \\
K &= \begin{bmatrix} 0,009 & 0,0002 & -0,0003 \\ 0,0001 & 0,005 & -0,0045 \\ -0,002 & 0,0003 & 0,0089 \end{bmatrix}
\end{aligned}$$

Setelah nilai penguat umpan balik optimal  $K$  didapatkan, maka nilai sinyal kontrol  $u$  optimal yang meminimalkan fungsi biaya dapat ditentukan melalui persamaan (3.10).

### 3.3.1 Perancangan Estimator

Diagram blok estimator (filter Kalman) seperti ditunjukkan pada gambar di bawah ini



**Gambar 3.2** Diagram Blok Filter Kalman

Dengan estimasi *state* dinotasikan oleh  $\hat{x}(t)$  dan estimasi output dinotasikan oleh  $\hat{y}(t)$ , maka persamaan estimator dapat ditulis sebagai:

$$\dot{\hat{x}}(t) = A\hat{x} + Bu + L(y - \hat{y}) \quad (3.11)$$

Dan persamaan estimasi output

$$\hat{y} = C\hat{x} \quad (3.12)$$

Untuk mendapatkan nilai penguatan  $L$ , dibutuhkan matriks  $Qn$  dan  $Rn$ , yang merepresentasikan gangguan/*noise* pada sistem dan *noise* pada pengukuran/sensor. Setelah itu matriks  $Qn$  dan  $Rn$  dimasukkan untuk menyelesaikan persamaan Riccati.

$$AP + PA^T - PC^T Rn^{-1} CP + Qn = 0$$

Pada tugas akhir ini, sistem akan diberi gangguan/*noise* sebesar 5% sampai 10% dari titik kerja. Untuk sistem dengan *noise* 5%, maka nilai matriks  $Qn$  dan  $Rn$  adalah

$$Qn = \begin{bmatrix} 10,8 & 0 & 0 \\ 0 & 6,665 & 0 \\ 0 & 0 & 0,01 \end{bmatrix}$$

$$Rn = \begin{bmatrix} 0,034 & 0 & 0 \\ 0 & 0,069 & 0 \\ 0 & 0 & 0,0433 \end{bmatrix}$$

Dari matriks  $Qn$  dan  $Rn$  maka matriks  $P$  bisa didapatkan penyelesaian persamaan Riccati

$$\begin{bmatrix} -0,0025 & 0 & 0 \\ 0,0694 & -0,1 & 0 \\ -0,0067 & 0 & 0 \end{bmatrix} P + P \begin{bmatrix} -0,0025 & 0,0694 & -0,0067 \\ 0 & -0,1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - P \begin{bmatrix} 1 & 0 & 0,0063 \\ 0 & 1 & 0 \\ 0 & 0 & 0,0047 \end{bmatrix} \begin{bmatrix} 29,41 & 0 & 0 \\ 0 & 14,49 & 0 \\ 0 & 0 & 23,09 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0,0063 & 0 & 0,0047 \end{bmatrix} P + \begin{bmatrix} 10,8 & 0 & 0 \\ 0 & 6,665 & 0 \\ 0 & 0 & 0,01 \end{bmatrix} = 0$$

Dengan menggunakan bantuan *software* Matlab maka didapatkan solusi dari persamaan Riccati, yaitu matriks  $P$  sebagai berikut:

$$P = \begin{bmatrix} 0,6059 & 0,0015 & -0,0003 \\ 0,0015 & 0,6710 & 0 \\ -0,0003 & 0 & 4,4281 \end{bmatrix}$$

Selanjutnya matriks  $P$  yang sudah didapatkan disubstitusikan ke dalam persamaan di bawah ini untuk mendapatkan nilai matriks penguat umpan balik optimal, yaitu  $L$ .

$$L = PC^T Rn^{-1} \quad (3.13)$$

$$L = \begin{bmatrix} 0,3029 & 0,0008 & -0,0002 \\ 0,0008 & 0,3331 & 0 \\ -0,0002 & 0 & 2,2115 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0,0063 \\ 0 & 1 & 0 \\ 0 & 0 & 0,0047 \end{bmatrix} \begin{bmatrix} 29,41 & 0 & 0 \\ 0 & 14,49 & 0 \\ 0 & 0 & 23,09 \end{bmatrix}$$

$$L = \begin{bmatrix} 17,81 & 0,02 & 0,08 \\ 0,044 & 9,72 & 0,0002 \\ -0,009 & 0 & 0,4806 \end{bmatrix}$$

Untuk sistem dengan *noise* 10%, maka nilai matriks  $Qn$  dan  $Rn$  adalah

$$Qn = \begin{bmatrix} 5,4 & 0 & 0 \\ 0 & 3,33 & 0 \\ 0 & 0 & 0,005 \end{bmatrix}$$

$$Rn = \begin{bmatrix} 0,017 & 0 & 0 \\ 0 & 0,0345 & 0 \\ 0 & 0 & 0,0216 \end{bmatrix}$$

Dari matriks  $Qn$  dan  $Rn$  maka matriks  $P$  bisa didapatkan penyelesaian persamaan Riccati

$$\begin{bmatrix} -0,0025 & 0 & 0 \\ 0,0694 & -0,1 & 0 \\ -0,0067 & 0 & 0 \end{bmatrix} P + P \begin{bmatrix} -0,0025 & 0,0694 & -0,0067 \\ 0 & -0,1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - P \begin{bmatrix} 1 & 0 & 0,0063 \\ 0 & 1 & 0 \\ 0 & 0 & 0,0047 \end{bmatrix} \begin{bmatrix} 58,82 & 0 & 0 \\ 0 & 29,41 & 0 \\ 0 & 0 & 46,29 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0,0063 & 0 & 0,0047 \end{bmatrix} P + \begin{bmatrix} 5,4 & 0 & 0 \\ 0 & 3,33 & 0 \\ 0 & 0 & 0,005 \end{bmatrix} = 0$$

Dengan menggunakan bantuan *software* Matlab maka didapatkan solusi dari persamaan Riccati, yaitu matriks  $P$  sebagai berikut:

$$P = \begin{bmatrix} 0,3029 & 0,0008 & -0,0002 \\ 0,0008 & 0,3331 & 0 \\ -0,0002 & 0 & 2,2115 \end{bmatrix}$$

Selanjutnya matriks  $P$  yang sudah didapatkan disubstitusikan ke dalam persamaan di bawah ini untuk mendapatkan nilai matriks penguat umpan balik optimal, yaitu  $L$ .

$$L = PC^T Rn^{-1}$$

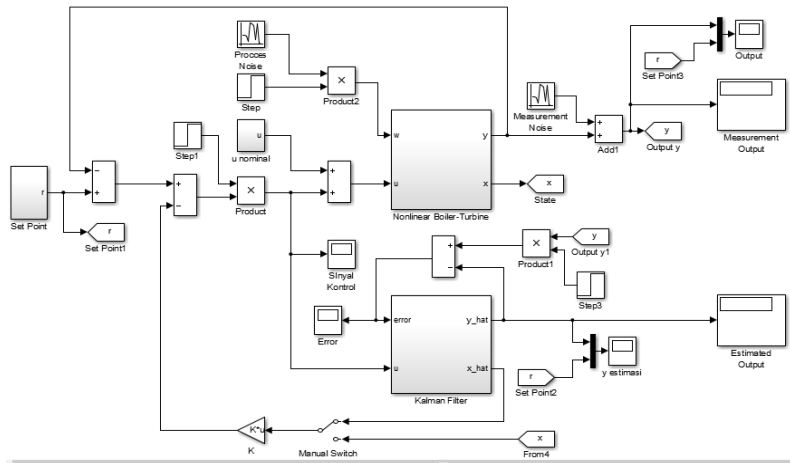
$$L = \begin{bmatrix} 0,3029 & 0,0008 & -0,0002 \\ 0,0008 & 0,3331 & 0 \\ -0,0002 & 0 & 2,2115 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0,0063 \\ 0 & 1 & 0 \\ 0 & 0 & 0,0047 \end{bmatrix} \begin{bmatrix} 58,82 & 0 & 0 \\ 0 & 29,41 & 0 \\ 0 & 0 & 46,29 \end{bmatrix}$$



$$L = \begin{bmatrix} 17,81 & 0,0223 & 0,0883 \\ 0,044 & 9,7971 & 0,0002 \\ -0,0097 & 0 & 0,4812 \end{bmatrix}$$

### 3.4 Perancangan Simulasi *Boiler-Turbine*

Hasil perancangan simulasi *boiler-turbine* menggunakan kontroler optimal *Linear Quadratic Gaussian* dengan menggunakan *software* Matlab adalah sebagai berikut:

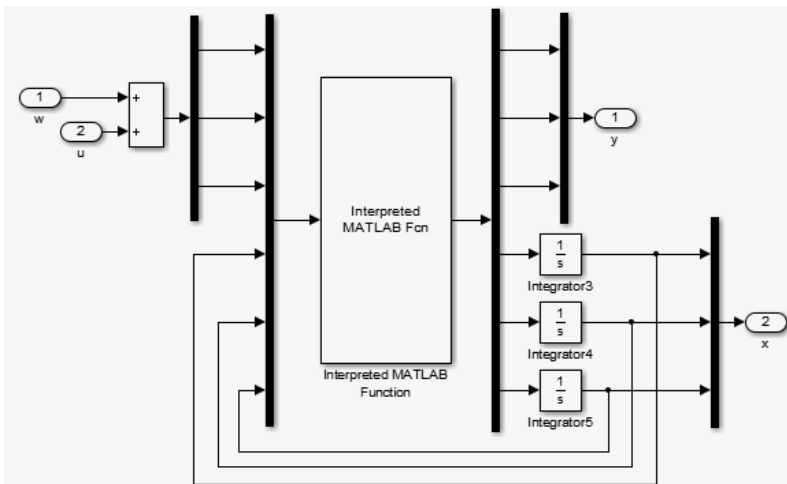


**Gambar 3.3** Diagram Blok Simulasi di Simulink

Pada perancangan simulasi *boiler-turbine* dengan kontroler *Linear Quadratic Gaussian* menggunakan *software* Matlab, kontroler LQG tidak diaktifkan sejak awal, namun kontroler dipasang ketika output sistem sudah mencapai keadaan *steady state*, yaitu sekitar pada detik ke 20.000. Sedangkan pada detik ke nol sampai sebelum 20.000, plant masih beroperasi secara manual tanpa adanya kontroler. Untuk mengaktifkan kontroler supaya hanya bekerja mulai detik ke 20.000 dan tidak aktif sebelum detik ke 20.000, maka digunakan fungsi *step* sebagai pengali untuk mengaktifkan kontroler. Pada fungsi *step*, diberi nilai *initial value* 0, *final value* 1, dan *step time* 20.000. Dengan demikian, kontroler hanya

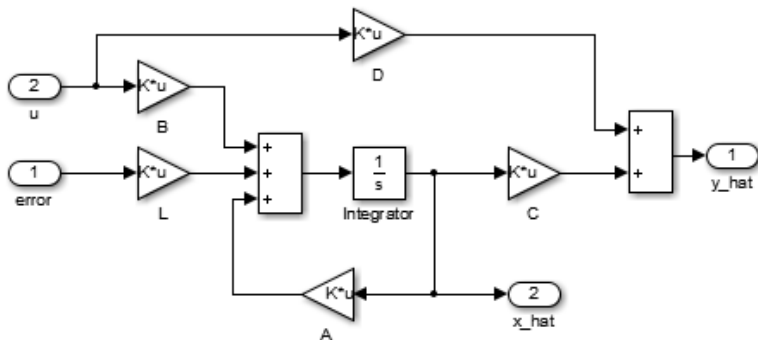
akan aktif mulai detik ke 20.000 dan sebelum detik ke 20.000 kontroler masih belum aktif.

Kemudian plant yang digunakan adalah plant aslinya, yaitu plant dengan model nonlinier. Sedangkan linierisasi digunakan hanya untuk mendesain kontroler *Linear Quadratic Gaussian*. Berikut adalah diagram dari subsistem plant *nonlinear boiler-turbine* di Simulink seperti pada gambar di bawah ini:



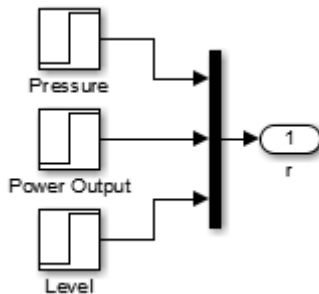
**Gambar 3.4** Diagram Blok Subsistem *Nonlinear Boiler-Turbine*

Estimator yang digunakan, yaitu filter Kalman, digunakan untuk mengestimasi variabel-variabel yang tak terukur. Parameter-parameter yang dipakai pada filter Kalman, yaitu matriks A, B, C, dan D didapatkan dari hasil linierisasi plant *nonlinear boiler-turbine*. Sedangkan matriks penguat umpan balik L (*gain kalman*) didapatkan dari perhitungan pada persamaan (3.11).



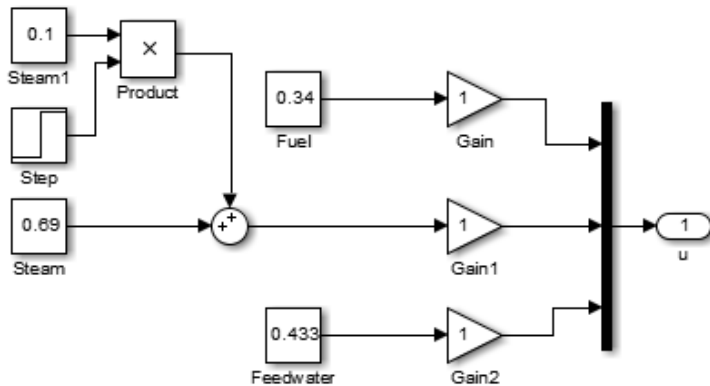
**Gambar 3.5** Diagram Blok Subsistem Filter Kalman

Subsistem dari *set point* adalah berisi nilai referensi, yaitu nilai output yang diinginkan. Pada sistem *nonlinear boiler-turbine*, terdiri dari tiga input dan tiga output. Output yang dikendalikan adalah tekanan, daya output, dan ketinggian air dalam tangki.



**Gambar 3.6** Diagram Blok Subsistem Sinyal Referensi

Subsistem *u* nominal berisi nilai input sistem pada titik kerja yang diinginkan, yaitu pada saat *boiler* mulai beroperasi. Pada saat *boiler* mulai dioperasikan, *boiler* beroperasi manual tanpa adanya kontroler, sehingga input hanya berasal dari *u* nominal saja. Kemudian setelah pada detik ke 20.000, kontroler mulai bekerja. Pada detik ke 21.000 plant diberi perubahan beban pada *steam* untuk menguji apakah kontroler bisa menangani perubahan beban yang terjadi atau tidak.



**Gambar 3.7** Diagram Blok Subsistem U Nominal

### 3.5 Perancangan *Human Machine Interface*

Virtual merupakan salah satu sarana penghubung antara manusia dengan mesin. Luaran (*output*) sistem perlu ditampilkan kepada pengguna berupa grafik animasi. Oleh karena itu perlu dirancang suatu *virtual boiler-turbine* melalui bantuan sebuah perangkat lunak (*software*). Untuk merancang sebuah *virtual boiler-turbine* digunakan *software* Wonderware Intouch. Tampilan *virtual plant boiler-turbine* yang telah dirancang melalui *software* Wonderware Intouch dapat dilihat pada Gambar 3.2.

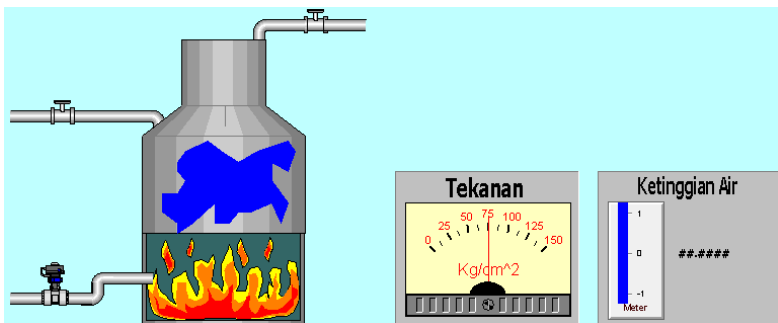
HMI (*Human Machine Interface*) adalah membuat fungsi dari teknologi nyata. Dengan membuat desain HMI yang sesuai, akan membuat pekerjaan fisik lebih mudah pada hampir semua solusi teknis, efektifitas dari HMI adalah dapat memprediksi penerimaan *user* terhadap seluruh solusi yang ada. Konsep HMI yang modern pada industri adalah sebagai media komunikasi antara operator dengan perancangan yang secara ideal mampu memberikan informasi yang diperlukan, agar perencanaan yang dilakukan dengan tingkat efisiensi maksimum. HMI merupakan sarana bagi operator untuk mengakses sistem otomatisasi di lapangan yang mencakup operasional, pengembangan, dan perawatan *troubleshooting*.

HMI biasa digunakan dalam dunia industri disebut juga sebuah tempat di mana interaksi antara manusia dan mesin terjadi. Tujuan dari interaksi antara manusia dan mesin pada antar muka pengguna adalah

pengoperasian dan kontrol mesin yang efektif, dan umpan balik dari mesin yang membantu operator dalam membuat keputusan operasional. Contoh-contoh dari konsep luas antarmuka pengguna ini termasuk aspek-aspek interaktif dari sistem operasi komputer, alat-alat, kontrol operator mesin berat, dan kontrol proses. Pertimbangan desain berlaku ketika membuat antarmuka pengguna yang berkaitan atau melibatkan disiplin-disiplin ilmu seperti ergonomi dan psikologi.

Antarmuka pengguna mencakup perangkat keras dan perangkat lunak. Antarmuka pengguna hadir untuk berbagai sistem, dan menyediakan cara untuk:

- ✓ Input, memungkinkan pengguna untuk memanipulasi sebuah sistem.
- ✓ Output, memungkinkan sistem untuk menunjukkan efek dari manipulasi pengguna.



**Gambar 3.8** Tampilan *Human Machine Interface*

*Halaman ini sengaja dikosongkan*

## **BAB IV**

### **PENGUJIAN DAN ANALISIS DATA**

Perancangan sistem yang telah dilakukan pada Bab 3, maka langkah selanjutnya adalah melakukan pengujian terhadap sistem perancangan dan *plant* dengan cara mengintegrasikan sistem. Pengujian dilakukan dengan menggunakan simulasi di MATLAB. Hasil dari simulasi kemudian di analisa dan apakah kontroler mampu mengikuti perubahan *plant* serta membandingkan terhadap model referensi yang diinginkan.

#### **4.1 Simulasi**

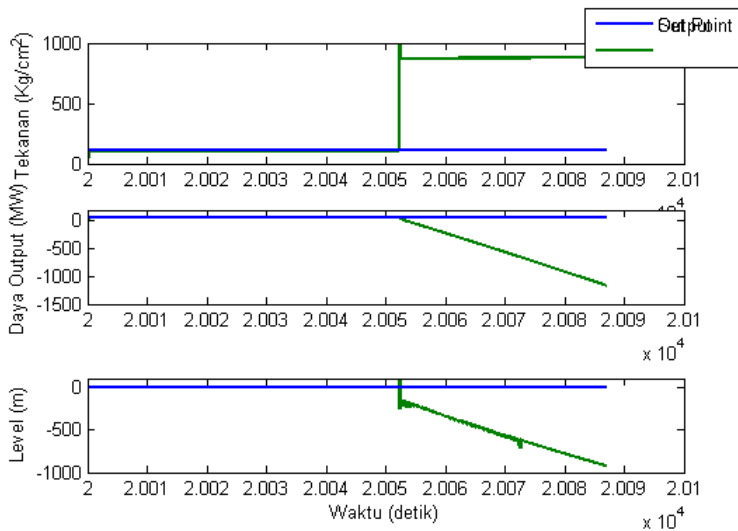
Setelah perancangan kontroler, maka selanjutnya untuk menguji apakah kontroler yang sudah dirancang bias bekerja dengan baik atau tidak, maka perlu dilakukan simulasi.

Kontroler diaplikasikan pada system aslinya, yaitu system yang nonlinier, dengan tujuan untuk menguji apakah kontroler bias mengatasi *noise* yang ada pada system, baik itu *noise* pada sinyal kontrol maupun *noise* pada sensor, juga untuk menguji apabila system mengalami perubahan beban.

Pada tugas akhir ini, simulasi dilakukan dengan menggunakan bantuan *software* Matlab R2013a.

##### **4.1.1 Simulasi Menggunakan LQR**

Sebelum dilakukan simulasi menggunakan kontroler LQG, perlu dilakukan simulasi menggunakan kontroler LQR dulu untuk mengetahui seberapa pengaruh *noise* pada sistem apabila dikontrol hanya dengan menggunakan penguat umpan balik LQR saja. Berikut hasil simulasi sistem menggunakan kontroler LQR dengan *noise* sebesar 5% dari titik kerja.



**Gambar 4.1** Output Sistem Dengan *Noise* 5% Menggunakan LQR

Dari simulasi menggunakan kontroler LQR di atas, terlihat bahwa sistem tidak bias mengatasi *noise* sehingga sistem menjadi semakin tidak stabil dan tidak bias dikendalikan.

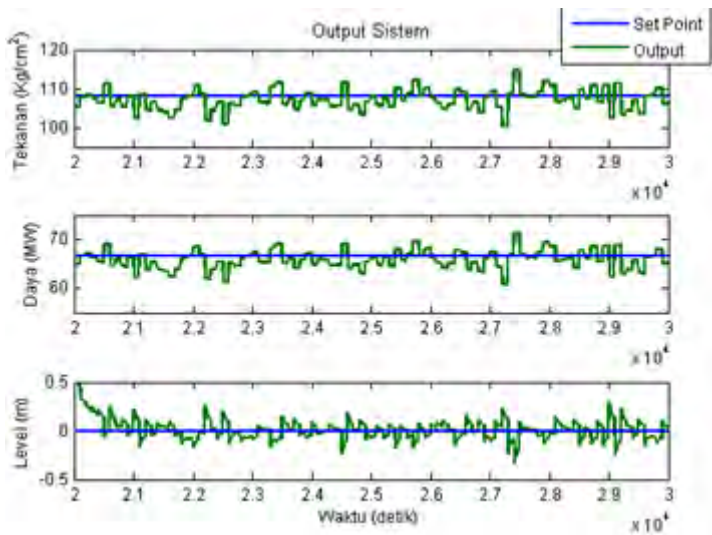
#### 4.1.2 Simulasi Menggunakan LQG

Pada simulasi menggunakan kontroler *Linear Quadratic Gaussian*, dilakukan simulasi dengan pemberian *noise* 5% sampai 10% dari titik kerja yang telah ditentukan. Untuk sistem dengan titik kerja

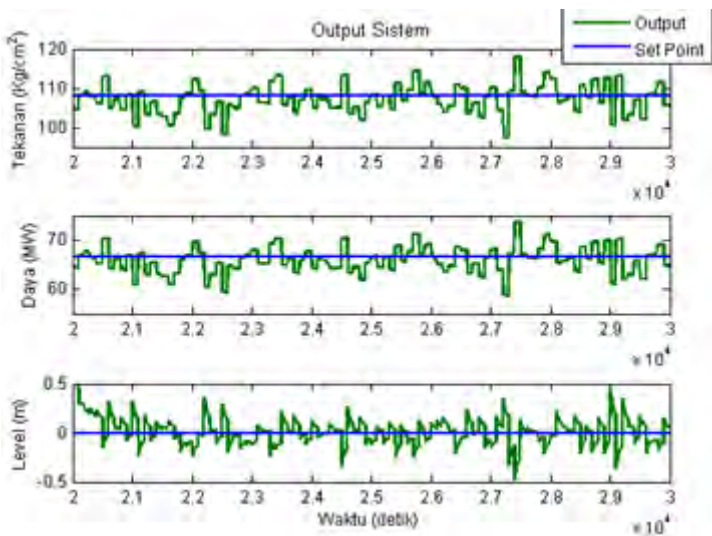
nominal, yaitu pada titik  $x = \begin{bmatrix} 108 \\ 66,65 \\ 428 \end{bmatrix}$ ,  $u = \begin{bmatrix} 0,34 \\ 0,69 \\ 0,433 \end{bmatrix}$ , dan  $y = \begin{bmatrix} 108 \\ 66,65 \\ 0 \end{bmatrix}$ .

Selain itu juga dilakukan simulasi apabila sistem mengalami perubahan beban. Berikut hasil simulasi menggunakan *software* Matlab dengan kontroler LQG ditampilkan pada gambar di bawah ini.





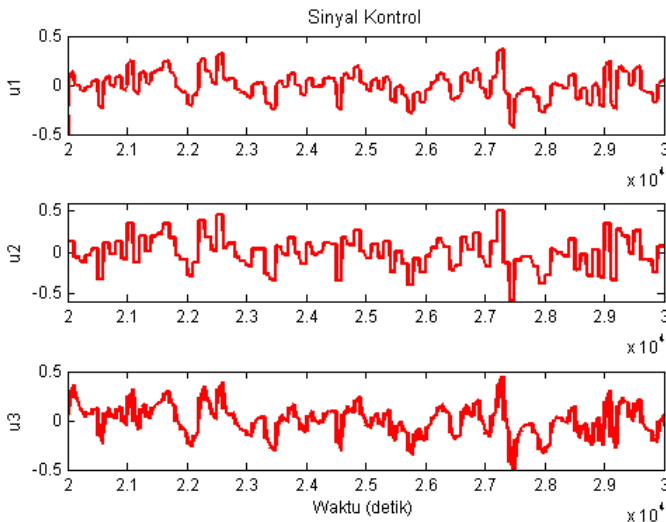
**Gambar 4.2** Output Sistem Dengan *Noise* 5% Dari Titik Kerja



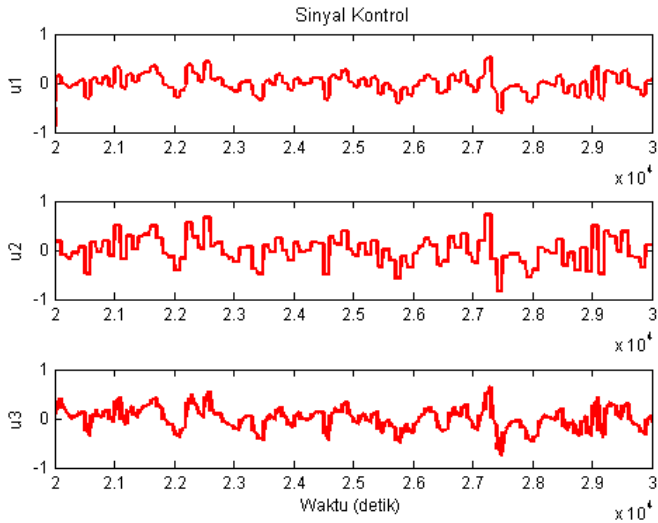
**Gambar 4.3** Output Sistem Dengan *Noise* 10% Dari Titik Kerja

Pada gambar hasil simulasi di atas, grafik warna hijau adalah output sistem, sedangkan grafik warna biru adalah *set point*. Dari grafik hasil simulasi menggunakan kontroler LQG di atas, dapat diketahui bahwa output sistem dengan menggunakan kontroler LQG masih bias bekerja di sekitar titik kerja yang diinginkan meskipun diberi *noise* sampai 10 persen dari titik kerja.

Untuk sinyal kontrol ketika sistem diberi *noise* sebesar 5%-10% dari titik kerja ditampilkan pada gambar 4.3 dan gambar 4.4.

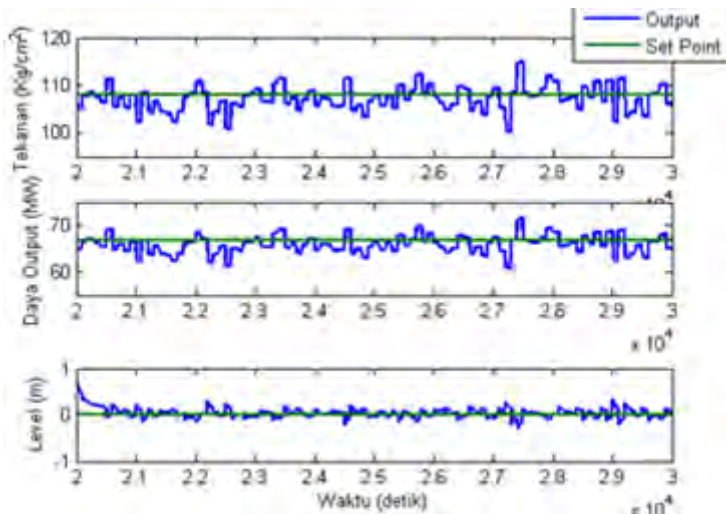


**Gambar 4.4** Sinyal Kontrol Untuk Sistem Dengan *Noise* 5% Dari titik Kerja

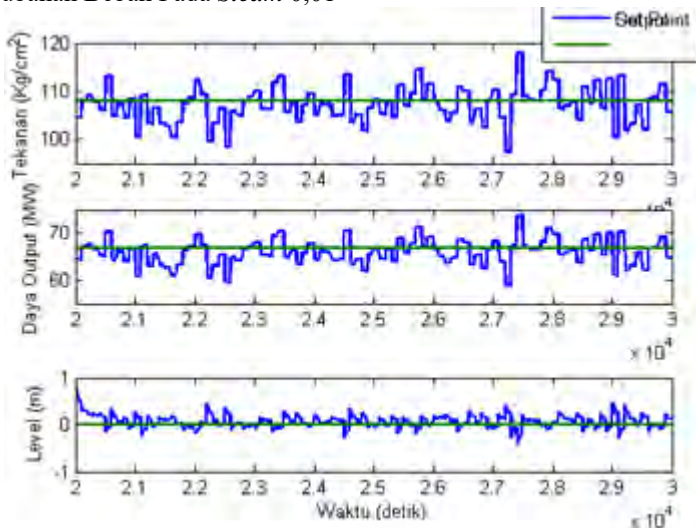


**Gambar 4.5** Sinyal Kontrol Untuk Sistem Dengan *Noise* 10% Dari titik Kerja

Sistem dengan kontroler LQG juga akan diuji dengan pemberian perubahan beban pada *steam* sebesar 0,01 dari titik kerja. Berikut hasil simulasi sistem menggunakan kontroler LQG apabila sistem diberi perubahan beban ditampilkan pada gambar di bawah ini.



**Gambar 4.6** Output Sistem Dengan *Noise* 5% Dari titik Kerja Dan Perubahan Beban Pada *Steam* 0,01



**Gambar 4.7** Output Sistem Dengan *Noise* 10% Dari titik Kerja Dan Perubahan Beban Pada *Steam* 0,01

Untuk sistem dengan perubahan beban pada *steam*, dari grafik di atas dapat diketahui bahwa kontroler LQG mampu mengatasi perubahan beban yang terjadi sehingga sistem tetap bekerja pada titik operasi yang diinginkan.

*Halaman ini sengaja dikosongkan*

## LAMPIRAN

### Program *Linear Quadratic Gaussian* pada Matlab

```
clc;
A = [-0.0025    0    0;
      0.0694 -0.1  0;
     -0.0067    0    0];

B = [0.9 -0.349 -0.15;
      0  14.16   0 ;
      0  -1.398  1.659];

C = [ 1    0    0 ;
      0    1    0 ;
      0.0063 0 0.0047];

D = [ 0    0    0 ;
      0    0    0 ;
      0.2533 0.5124 -0.014];

Q = [0.01  0    0 ;
      0  0.01  0 ;
      0    0  0.01];

R = [100 0 0 ;
      0 100 0 ;
      0 0 100];

Qa = [10.8  0    0 ;
       0  6.66  0 ;
       0    0  0.01];

Qa1 = [5.4  0    0 ;
        0  3.33  0 ;
        0    0  0.005];

Ra = [0.034  0    0 ;
       0  0.069  0 ;
       0    0  0.0433];

Ra1 = [0.017  0    0 ;
        0  0.034  0 ;
```

```

          0      0  0.0216];
G = [1 0 0;
     0 1 0;
     0 0 1];
ob = obsv(A,C)
unob = length(A)-rank(ob)    %mengecek
observabilitas
co = ctrb(A,B)
unco = length(A)-rank(co)    %mengecek
controlabilitas
[S,a,K] = care(A,B,Q,R);
[X,m,L] = care(A',C',G*Qa*G',Ra);
[Xl,m1,L1] = care(A',C',G*Qa1*G',Ra1);
eig(A-B*K)                   %mengecek stabilitas
plant
eig(A'-C'*L)                 %mengecek stabilitas
filter Kalman
eig(A'-C'*L1)                %mengecek stabilitas
filter Kalman

```





## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Berdasarkan pengujian yang telah dilakukan selama tugas akhir ini, maka dapat diambil beberapa kesimpulan yaitu:

- ✓ Kontroler LQR tidak mampu menangani apabila sistem diberi *noise*
- ✓ Kontroler LQG mampu bekerja dengan baik meskipun sistem diberi *noise* mulai 5% sampai 10% dari titik kerja nominal
- ✓ Kontroler LQG juga mampu mengatasi perubahan beban yang terjadi pada *steam*. Output sistem tetap bekerja dengan baik, yaitu di sekitar nilai set point yang diinginkan.

#### **5.2 Saran**

Pada riset yang akan datang, diharapkan kontroler yang digunakan adalah kontroler nonlinier karena sistem merupakan sistem yang nonlinier dengan sifat ke-nonlinier-an yang tinggi sehingga sangat sulit untuk dikendalikan.

## DAFTAR PUSTAKA

- [1]. Wahyudianto, D.K., “*Desain Sistem Kontrol Menggunakan Fuzzy Gain Scheduling Untuk Unit Boiler-Turbine Nonlinear*”, Tugas Akhir, Teknik Elektro FTI-ITS Surabaya, 2013.
- [2]. Bell, R.D. and Åström, K.J., “*Dynamic Models for Boiler- Turbine- Alternator Units: Data Logs and Parameter Estimation for a 160 MW Unit*”. Report TFRT-3192, Lund Institute of Technology, Sweden, 1987.
- [3]. Tan, W., Horacio, J.M., Chen, T., and Liu, J., “*Analysis and Control of a Nonlinear Boiler-Turbine Unit*”, Journal of Process Control 15, pp. 883–891, 2005.
- [4]. Lu, Z., Wei, L., Gang, F., Feng, W., “*A Study of Nonlinear Control Schemes for A Boiler-Turbine Unit*”, 2010.
- [5]. Vanessa Sarah Griselda, “*Kontrol Optimal Berdasarkan Pada Kalkulus Variasi*,” Tugas Akhir, Institut Teknologi Bandung, Bandung, 2010.
- [6]. Ogata, Katsuhiko, “*Modern Control Engineering 5<sup>th</sup> Ed.*”, 2010.
- [7]. Lewis, Frank, “*Optimal Control 3<sup>rd</sup> Ed.*”, 2012.

## RIWAYAT PENULIS



**Yudi Galih Permana** lahir di Sidoarjo 16 Desember 1993. Menyelesaikan sekolah di SD Negeri 2 Kedungbocok, Sidoarjo pada tahun 2006. Melanjutkan ke SMP Negeri 2 Tarik, Sidoarjo dan lulus pada tahun 2009. Kemudian melanjutkan ke SMA Negeri 1 Puri, Mojokerto dan lulus pada tahun 2012. Karena penulis memiliki minat pada bidang teknik khususnya bidang teknik elektro, penulis melanjutkan studi ke Institut Teknologi Sepuluh Nopember Surabaya Jurusan Teknik Elektro. Pada semester 5, penulis mulai menekuni salah satu bidang studi yang ada di Jurusan Teknik Elektro ITS, yaitu Teknik Sistem Pengaturan. Pada bulan Juni 2016, penulis mengikuti seminar dan ujian Tugas Akhir di Bidang Studi Teknik Sistem Pengaturan Jurusan Teknik Elektro FTI-ITS, Surabaya sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro dari Institut Teknologi Sepuluh Nopember Surabaya.